EXPERIMENT NO.

1 PROGRAM:

Name: Iqra Farid

Roll no:25

Date of performance:02-08-24 Date of submission:09-08-24

```c
#include <stdio.h>
#include<conio.h>

int STK[100], top= -1, i, n, x, ch;
void Push();
void Pop();
void Peep();
void Display();

void main()

{
   printf("\n Implementation of STACK using array \n");
   printf("Enter size of stack (Maximum size = 100): ");
   scanf("%d", &n);

   do
   {
      printf("\n Stack Operations: \n");
      printf("\n1.Push\n 2.Pop\n 3.Peep\n4. Display\n 5.Exit \n");
      printf("\n Enter your choice: ");
      scanf("%d", &ch);
      switch (ch)
      {
      case 1:
         Push();
         break;
      case 2:
         Pop();
         break;
      case 3:
         Peep();
         break;
      case 4:
         Display();
         break;
      case 5:
         printf("Exit: Program Finished  ");
         break;
      default:
         printf("Please enter a valid choide: 1, 2, 3, 4, 5 \n");
      }
   } while (ch != 5);
}

// PUSH FUNCTION
void Push()
```

```
    }
}

// PEEP FUNCTION
void Peep()
{
    printf(" Enter the position of the element
from the top which you want to peep: ");
    scanf("%d", &i);
    if (top - i + 1 < 0)
    {
        printf(" Stack Underflow on Peep \n");
    }
    else
    {
        printf(" The %d element from the top is: %d
\n", i, STK[top- i + 1]);
    }
}

//  DISPLAY FUNCTION
void Display()
{
    if (top < 0)
    {
        printf(" Stack is empty \n");
    }
    else
    {
        printf(" The element in the stack are:");
        for (i =top; i > -1; i--)
        {
            printf("\n %d \n", STK[i]);
        }
    }
}
```

```
    } while (ch != 5);
}

//  PUSH FUNCTION
void Push()
{
    if (top >= n - 1)
    {
        printf(" Stack Overflow \n");
    }
    else
    {
        printf(" Enter the element to be pushed: ");
        scanf("%d", &x);
        top++;
        STK[top] = x;
    }
}

// POP FUNCTION
void Pop()
{
    if (top < 0)
    {
        printf(" Stack Underflow \n");
    }
    else
    {
        printf(" The popped element is: %d \n",
STK[top]);
        top--;
    }
}

```

OUTPUT:

```
4.Display
5.Exit

 Enter your choice: 1
 Enter the element to be pushed: 3

 Stack Operations:

1.Push
2.Pop
3.Peep
4.Display
5.Exit

 Enter your choice: 1
 Enter the element to be pushed: 4

 Stack Operations:

1.Push
2.Pop
3.Peep
4.Display
5.Exit

 Enter your choice: 2
 The popped element is: 4

 Stack Operations:

1.Push
2.Pop
3.Peep
4.Display
5.Exit

 Enter your choice: 3
 Enter the position of the element from the top which you want to p
eep: 1
 The 1 element from the top is: 3

 Stack Operations:

1.Push
2.Pop
3.Peep
4.Display
5.Exit

 Enter your choice: 4
 The element in the stack are:
 3

 Stack Operations:

1.Push
2.Pop
3.Peep
4.Display
5.Exit

 Enter your choice: 5
Exit: Program Finished
[Program finished]
```

```
 Implementation of STACK using array
Enter size of stack (Maximum size = 100): 5

 Stack Operations:

1.Push
2.Pop
3.Peep
4.Display
5.Exit

 Enter your choice: 1
 Enter the element to be pushed: 3

 Stack Operations:

1.Push
2.Pop
3.Peep
4.Display
5.Exit

 Enter your choice: 1
 Enter the element to be pushed: 4

 Stack Operations:

1.Push
2.Pop
3.Peep
4.Display
5.Exit

 Enter your choice: 2
 The popped element is: 4

 Stack Operations:

1.Push
2.Pop
3.Peep
4.Display
5.Exit

 Enter your choice: 3
 Enter the position of the element from the top which you want to p
eep: 1
 The 1 element from the top is: 3

 Stack Operations:

1.Push
2.Pop
3.Peep
4.Display
5.Exit

 Enter your choice: 4
 The element in the stack are:
 3

 Stack Operations:
```